UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/873,038	06/02/2001	Sanjay Lal	004906.P080	4240

8791 7590 03/02/2005

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
PATEL, HARESH N	

ART UNIT	PAPER NUMBER
2154	

DATE MAILED: 03/02/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/873,038	LAL, SANJAY	
	Examiner	Art Unit	
	Haresh Patel	2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11/19/2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-32* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-32* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☒ Interview Summary (PTO-413) Paper No(s)/Mail Date. *2/13/2005*.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-32 are presented for examination.

Response to Arguments

2. Applicant's arguments filed 11/19/04 have been fully considered but they are not persuasive.

Applicant argues (1) combined teachings of Kranich et al, 6,651,163 (Hereinafter Kranich) and Yoshioka et al, 6,425,039 (Hereinafter Yoshika) do not disclose, "executing a number of instructions at an address within a common interrupt handling vector address space, wherein the number of instructions cause the processor to determine an identification of the processor based on a query that is internal to the processor". The examiner disagrees in response to applicant's arguments. Yoshioka teaches executing a number of instructions at an address within a common interrupt handling vector address space of the same memory (e.g., concept of handling common exception events, figures 2, 5 and 13). Kranich teaches a processor that processes queries related to both internal and external to the processor in order to help determine the identification of the processor (e.g., use of internal and external registers, col., 1, line 13 – col., 2, line 33). Teachings of Yoshioka, i.e., to handle an exception in a multiprocessor environment using the common exception handling vector would help handle the exception using the processor. The query within the processor would help identify the processor in order to handle the exception. The limitations, "executing a number of instructions at an address within a common interrupt handling vector address space, wherein the number of instructions cause the processor to determine an identification of the processor based on a query that is internal to the

processor", is also addressed by the new ground(s) of rejection (please refer to the below rejections of this office action). Therefore, examiner believes that the claimed limitations are taught by the prior art.

Applicant argues (2) Kranich does not disclose, "the execution of an instruction during exception handling that requires a processor to check its identification". The examiner disagrees in response to applicant's arguments. Kranich teaches the execution of an instruction during exception handling that requires a processor to check its identification (e.g., use of internal and external registers to determine the processor number, figure 11, col., 1, line 13 – col., 2, line 33). The limitations, "the execution of an instruction during exception handling that requires a processor to check its identification", is also addressed by the new ground(s) of rejection (please refer to the below rejections of this office action). Therefore, examiner believes that the claimed limitations are taught by the prior art.

Applicant argues, (3) "Yoshioka and Kranich are improperly combined". The examiner disagrees in response to applicant's arguments. Both the cited references Yoshika and Kranich teach what the applicant is trying to accomplish, i.e., the claimed invention by the usage of preambles "a method comprising", "a system comprising", "the machine to perform operations comprising", and, "a method for handling a number of execeptions within a processor in a multi-procesing system". Kranich teaches a processor that processes queries related to both internal and external to the processor in order to help determine the identification of the processor (e.g., use of internal and external registers, col., 1, line 13 – col., 2, line 33). Yoshioka teaches executing a number of instructions at an address within a common interrupt handling vector address space of the same memory (e.g., concept of handling common exception events, figures

2, 5 and 13). The motivation would be obvious because the teachings of Yoshioka, i.e., to handle an exception in a multiprocessor environment using the common exception handling vector would help handle the exception using the processor. The query within the processor would help identify the processor in order to handle the exception. Also, The test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of a primary reference. It is also not that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinally skill in the art. In re Keller, 642 F.2d 414, 425, 208 USPQ 871, 881 (CCPA 1981); In re Young, 927 F.2d 588, 591, 18 USPQ2d 1089, 1091 (Fed. Cir. 1991). Therefore, Yoshika and Kranich meet the claimed limiations.

Applicant argues (4) combined teachings of Kranich and Yoshioka do not disclose, "executing a number of instructions at an address within the common interrupt handling address space of the same memory, wherein a number of instructions cause a processor to read a bit within an internal register to determine an identification of a processor and a multiprocessor system". The examiner disagrees in response to applicant's arguments. Yoshioka teaches executing a number of instructions at an address within a common interrupt handling vector address space of the same memory (e.g., concept of handling common exception events, figures 2, 5 and 13). Kranich teaches wherein a number of instructions cause a processor to read a bit within an internal register to determine an identification of a processor and a multiprocessor system (e.g., use of bit information of internal and external registers to determine the processor number, figure 11, col., 1, line 13 – col., 2, line 33). Teachings of Yoshioka, i.e., to handle an exception in a multiprocessor environment using the common exception handling vector would

help handle the exception using the processor. The query, i.e., read a bit within an internal register within the processor would help identify the processor in order to handle the exception. The limitations, "executing a number of instructions at an address within the common interrupt handling address space of the same memory, wherein a number of instructions cause a processor to read a bit within an internal register to determine an identification of a processor and a multiprocessor system", is also addressed by the new ground(s) of rejection (please refer to the below rejections of this office action). Therefore, examiner believes that the claimed limitations are taught by the prior art.

Applicant argues (5) combined teachings of Brenner, Jr., et al., U.S. publication 2002/00713154 (Hereinafter Brenner) and Browning et al, 6,006,247 (Hereinafter Browning) do not disclose, "executing a number of instructions at an address within a common interrupt handling vector address space, wherein the number of instructions cause the processor to determine an identification of the processor based on a query that is internal to the processor". The examiner disagrees in response to applicant's arguments. Brenner teaches executing a number of instructions at an address within a common interrupt handling vector address space of the same memory (e.g., concept of handling common exception/interrupt events, figure 2). Browning teaches a processor that processes queries related to both internal and external to the processor in order to help determine the identification of the processor (e.g., use of internal and external registers, figure 2). Teachings of Brenner, i.e., to handle an exception in a multiprocessor environment using the common exception handling vector would help handle the exception using the processor. The query within the processor would help identify the processor in order to handle the exception. The limitations, "executing a number of instructions at an

address within a common interrupt handling vector address space, wherein the number of instructions cause the processor to determine an identification of the processor based on a query that is internal to the processor", is also addressed by the new ground(s) of rejection (please refer to the below rejections of this office action). Therefore, examiner believes that the claimed limitations are taught by the prior art.

Applicant argues (6) combined teachings of Brenner and Browning do not disclose, "executing a number of instructions at an address within the common interrupt handling address space of the same memory, wherein a number of instructions cause a processor to read a bit within an internal register to determine an identification of a processor and a multiprocessor system". The examiner disagrees in response to applicant's arguments. Brenner teaches executing a number of instructions at an address within a common interrupt handling vector address space of the same memory (e.g., concept of handling common exception events, figure 2). Browning teaches wherein a number of instructions cause a processor to read a bit within an internal register to determine an identification of a processor and a multiprocessor system (e.g., use of bit information of internal and external registers to determine the processor number, figure 11, col., 1, line 13 – col., 2, line 33). Teachings of Brenner, i.e., to handle an exception in a multiprocessor environment using the common exception handling vector would help handle the exception using the processor. The query, i.e., read a bit within an internal register within the processor would help identify the processor in order to handle the exception. The limitations, "executing a number of instructions at an address within the common interrupt handling address space of the same memory, wherein a number of instructions cause a processor to read a bit within an internal register to determine an identification of a processor and a multiprocessor

system", is also addressed by the new ground(s) of rejection (please refer to the below rejections of this office action). Therefore, examiner believes that the claimed limitations are taught by the prior art.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter, which the applicant regards as his invention.

3. Claims 1, 4-9, 11, 21, 24-29, 31 and 32 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

4. Claim 1, recites the limitation "the processor to determine an identification of the processor based on a query that is internal to the processor". There is insufficient antecedent basis for this limitation in the claim. Since, multiple processors exits in the claim, it is not clear which processor is referred by these limitations.

5. Claims 4, 5, recite the limitation "the processor". There is insufficient antecedent basis for this limitation in the claim. Since, multiple processors exits in the claim, it is not clear which processor is referred by these limitations.

6. Claims 6, 9, 11, 24, 25, 26, 29, 31, recite the limitation "the processor", "the processors", "the identification of the processor". There is insufficient antecedent basis for this limitation in the claim. Since, multiple different processors exits in the claim, it is not clear which processor / processors is referred by these limitations.

7. Claims 7, 27, recite the limitation "the different processor". There is insufficient antecedent basis for this limitation in the claim.

8. Claim 8, recites the limitation "the different processors", "the type of exception received within the processor", "the processor to read a bit within an internal register to determine an identification of the processor". There is insufficient antecedent basis for this limitation in the claim. Since, multiple processors exits in the claim, it is not clear which processor is referred by these limitations.

9. Claims 21, 28, recite the limitation "the different processors", "the type of exception received within the processor", "the processor to read a bit within an internal register to determine an identification of the processor". There is insufficient antecedent basis for this limitation in the claim. Since, multiple processors exits in the claim, it is not clear which processor is referred by these limitations.

10. Claim 32, recite the limitation "the processor determines a type of the exception", "the processor based on a value", "the operating system of the identified processor". There is insufficient antecedent basis for this limitation in the claim. Since, multiple processors, multiple exceptions and multiple operating systems exits in the claim, it is not clear which processor, operating system and exception is referred by these limitations.

Claim Rejections - 35 USC § 103

11. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

12. Claims 1-3, 21-23, are rejected under U.S.C. 103(a) as being unpatentable over Browning et al., 6,006,247, IBM (Hereinafter Browning-IBM) in view of O'Shea-Intel 6,611,911, Intel (Hereinafter O'Shea-Intel) and Endo et al., 6,615,303, Hitachi, Ltd, (Hereinafter Endo-Hitachi).

13. As per claims 1, 21, Browning-IBM clearly teaches the following:

a method and a machine-readable medium that provides instructions for handling a number of exceptions within a processor in a multi-processing system (e.g., col., 2, lines 40 –65, col., 4, lines 6 – 22, figure 1), comprising:

receiving an exception within the processor (e.g., col., 2, lines 40 –65), wherein each processor in the multi- processor system shares a same memory (e.g., figure 2, col., 4, lines 6 – 22);

executing a number of instructions at an address (e.g., col., 3, lines 46 – 61, col., 1, lines 36 – 52) within a common interrupt handling vector address space of the same memory (e.g., col., 3, line 65 – col., 4, line 29), wherein the number of instructions cause the processor to determine an identification of the processor (e.g., col., 4, lines 35 – 46),

modifying execution flow of the exception (e.g., col., 6, lines 16 – 36).

However, Browning-IBM does not specifically mention about a query that is internal to the processor.

O'Shea-Intel teaches a query that is internal to the processor (e.g., col., 4, lines 1 – 18).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM with the teachings of O'Shea-Intel in order to facilitate a query that is internal to the processor because the query would help get the

information from the processor. The well-known concept of using a query within the processor would help provide processor information when the query is performed.

Browning-IBM and O'Shea-Intel do not specifically mention about executing an interrupt handler located within one of a number of different interrupt handling vector address spaces.

Endo-Hitachi teaches executing an interrupt handler located within one of a number of different interrupt handling vector address spaces (e.g., figure 1).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM and O'Shea-Intel with the teachings of Endo-Hitachi in order to facilitate executing an interrupt handler located within one of a number of different interrupt handling vector address spaces because the interrupt handler would help handle the interrupt. The well-known concept of using different interrupt handlers stored in the memory would help handle different types of interrupts. The processor would help process the exceptions / interrupts using the different interrupt handlers.

14. As per claims 2, 3, 22, 23, Browning-IBM and O'Shea-Intel teach the claimed limitations as rejected above. However, Browning-IBM and O'Shea-Intel do not specifically mention about each processor in the multi-processor system executes one of a number of operating systems associated with one of the number of different interrupt handling vector address spaces.

Endo-Hitachi teaches each processor in the multi-processor system executes one of a number of operating systems (e.g., figure 1) associated with one of the number of different interrupt handling vector address spaces (e.g., figure 1).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM and O'Shea-Intel with the teachings of Endo-Hitachi in order to facilitate each processor in the multi-processor system to exectue one of a number of operating systems associated with one of the number of different interrupt handling vector address spaces because the processor would be able to use any operating system from the available multiple operating systems. Depending upon the operating system used by the processor would help the functionality provided by the operating system. The well-known concept of using associated interrupt handling space would help process the interrupt using the operating system handled by the processor.

15. Claims 4-5, 24-25, are rejected under U.S.C. 103(a) as being unpatentable over Browning-IBM, O'Shea-Intel, Endo-Hitachi and Rose, 6,314,500 IBM (Hereinafter Rose-IBM).

16. As per claims 4-5, 24-25, Browning-IBM, O'Shea-Intel and Endo-Hitachi teach the claimed limitations as rejected above. However, Browning-IBM, O'Shea-Intel and Endo-Hitachi do not specifically mention about reading a bit within an internal register and the register is not dedicated to determining the identification of the processor.

Rose-IBM teaches reading a bit within an internal register (e.g., col., 9, lines52 – 67) and the register is not dedicated to determining the identification of the processor (e.g., col., 9, lines 52 – 67).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel and Endo-Hitachi with the teachings of Rose-IBM in order to facilitate reading a bit within an internal register and the

register is not dedicated to determining the identification of the processor because a bit within a register would provide information of the processor among the group of processors. The well-known concept of using a register not dedicated for determining the identification of the processor would help the register to be used for determining other information besides only determining the identification of the processor.

17. Claims 6, 7, 26, 27, are rejected under U.S.C. 103(a) as being unpatentable over Browning-IBM, O'Shea-Intel, Endo-Hitachi in view of Colley et al., 5,113,523, NCUBE Corporation (Hereinafter Colley - NCUBE) and Stracovsky et al., 6,539,440, Infineon (Hereinafter Stracovsky-Infineon).

18. As per claims 6, 7, 26, 27, Browning-IBM, O'Shea-Intel and Endo-Hitachi teach the claimed limitations as rejected above. Browning-IBM also teaches that each of the number of exceptions received by the different processors (e.g., figure 2, col., 2, lines 44 – 64) execute instructions at an address within the common interrupt handling vector address space of the same memory (e.g., figure 2, col., 4, lines 6 - 45).

However, Browning-IBM, O'Shea-Intel and Endo-Hitachi do not specifically mention about determining the identification of the processor during initialization of the processor.

Colley – NCUBE teaches determining the identification of the processor during initialization of the processor (e.g., col., 27, lines 13 – 28).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel and Endo-Hitachi with the teachings of Colley – NCUBE in order to facilitate determining the identification of the

processor during initialization of the processor because the identification of the processor would help select the processor for processing information. The well-known concept of using a processor for initialization would help initialize the processor (itself) and the system.

Browning-IBM, O'Shea-Intel, Endo-Hitachi and Colley – NCUBE do not specifically mention about communications with a memory controller that is coupled between the processors in the multiple processor system and the same memory.

Stracovsky-Infineon teaches communications with a memory controller that is coupled between the processors in the multiple processor system and the same memory (e.g., col., 2, lines 17 – 36; col., 5, line 63 – col., 6, line 9).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel, Endo-Hitachi and Colley – NCUBE with the teachings of Stracovsky-Infineon in order to facilitate communications with a memory controller that is coupled between the processors in the multiple processor system and the same memory because the memory controller would help the processor to access the shared memory. The well-known concept of a memory usage would help provide information to the processors through the memory controller.

19. Claims 12 and 13 are rejected under U.S.C. 103(a) as being unpatentable over Browning-IBM in view of O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon and Nota et. al, 5,805,790, Hitachi Ltd (Hereinafter Nota-Hitachi).

20. As per claim 12, Browning-IBM clearly teaches the following:

a system / method / computer medium (e.g., col., 2, lines 40 –65, col., 4, lines 6 – 22, figure 1), comprising:

a memory (e.g., figure 2) that includes:

a common exception handling vector address space (e.g., col., 3, line 65 – col., 4, line 29, figure 2),

exception handling vector address space (e.g., col., 3, line 65 – col., 4, line 29, figure 2),

a first processor (e.g., col., 3, line 65 – col., 4, line 29, figure 2),

a second processor (e.g., col., 3, line 65 – col., 4, line 29, figure 2),

the second processor to execute a number of instructions in the common exception handling vector address space upon receipt of an exception (e.g., col., 4, lines 35 – 46, figure 2),

wherein the first processor and the second processor share the memory (e.g., col., 4, lines 15 – 46, figure 2),

wherein the number of instructions cause the second processor to determine an identification of the second processor (e.g., col., 4, lines 35 – 46).

However, Browning-IBM does not specifically mention about a query that is internal to the second processor.

O'Shea-Intel teaches a query that is internal to the processor (e.g., col., 4, lines 1 – 18).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM with the teachings of O'Shea-Intel in order to facilitate a query that is internal to the processor because the query would help get the

information from the processor. The well-known concept of using a query within the processor would help provide processor information when the query is performed.

Browning-IBM and O'Shea-Intel do not specifically mention about executing a number of exception handling vector address spaces, determining the type of exception received within the processor, and the processor determines a type of the exception.

Endo-Hitachi teaches executing a number of exception handling vector address spaces and a separate interrupt handler for each operating system (e.g., figure 1),

determining the type of exception received within the processor (e.g., figures 6 and 7), the processor determines a type of the exception (e.g., figures 6 and 7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM and O'Shea-Intel with the teachings of Endo-Hitachi in order to facilitate executing a number of exception handling vector address spaces because multiple exception handling vector address spaces would help handle multiple different interrupts/exceptions. The well-known concept of using different interrupt/exception handlers stored in the memory would help handle different types of interrupts/exceptions. The processor would help process the interrupts/exceptions using the different interrupt handlers based on determination of the interrupt/exception type.

Browning-IBM, O'Shea-Intel and Endo-Hitachi do not specifically mention about a memory controller that is coupled to the memory and the first and the second processors.

Stracovsky-Infineon teaches communications with a memory controller that is coupled to the memory and the first and the second processors (e.g., col., 2, lines 17 – 36; col., 5, line 63 – col., 6, line 9).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel and Endo-Hitachi with the teachings of Stracovsky-Infineon in order to facilitate a memory controller that is coupled to the memory and the first and the second processors because the memory controller would help the processors to access the shared memory. The well-known concept of a memory usage would help provide information to the processors through the memory controller.

Browning-IBM, O'Shea-Intel, Endo-Hitachi and Stracovsky-Infineon do not specifically mention about the first processor is to execute a first operating system and the second processor is to execute a second operating system.

Nota-Hitachi teaches the first processor is to execute a first operating system (e.g., col., 2, lines 24 – 29, figure 1) and the second processor is to execute a second operating system (e.g., col., 2, lines 24 – 29, figure 1).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel, Endo-Hitachi and Stracovsky-Infineon with the teachings of Nota-Hitachi in order to facilitate the first processor is to execute a first operating system and the second processor is to execute a second operating system because the first processor would help support the first operating system and the second processor would help support the second operating system. Having multiple processors would help support multiple operating systems. Having multiple different operating systems would help support functionality of all different operating systems supported by the system.

21. As per claim 13, Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon and Nota-Hitachi teach the claimed limitations as rejected above. Endo-Hitachi also teaches executing different operating systems associated with one of the number of different interrupt handling vector address spaces (e.g., figure 1).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel, Stracovsky-Infineon and Nota-Hitachi with the teachings of Endo-Hitachi in order to facilitate executing different operating systems associated with one of the number of different interrupt handling vector address spaces because different interrupt handlers would help handle interrupts for different operating systems. The well-known concept of using different interrupt handlers stored in the memory would help handle different types of interrupts depending upon the operating system used. The processor would help process the exceptions / interrupts using the different interrupt handlers.

22. Claims 8-10, 14, 15, 17-19, 28-30, 32 are rejected under U.S.C. 103(a) as being unpatentable over Browning-IBM in view of O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon, Nota-Hitachi and Rose-IBM.

23. As per claims 8-10, 14, 15, 17-19, 28-30, 32, Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon and Nota-Hitachi teach the claimed limitations as rejected above. However, Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon and Nota-Hitachi do not specifically mention about reading a bit within an internal register and the register is not dedicated to determining the identification of the processor.

Rose-IBM teaches reading a bit within an internal register (e.g., col., 9, lines52 – 67) and the register is not dedicated to determining the identification of the processor (e.g., col., 9, lines 52 – 67).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon and Nota-Hitachi with the teachings of Rose-IBM in order to facilitate reading a bit within an internal register and the register is not dedicated to determining the identification of the processor because a bit within a register would provide information of the processor among the group of processors. The well-known concept of using a register not dedicated for determining the identification of the processor would help the register to be used for determining other information besides only determining the identification of the processor.

24. Claims 11, 16, 20 and 31 are rejected under U.S.C. 103(a) as being unpatentable over Browning-IBM in view of O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon, Nota-Hitachi, Rose-IBM and Colley – NCUBE.

25. As per claims 11, 16, 20 and 31, Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon, Rose-IBM and Nota-Hitachi teach the claimed limitations as rejected above.

However, Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon, Rose-IBM and Nota-Hitachi do not specifically mention about determining the identification of the processor during initialization of the processor.

Colley – NCUBE teaches determining the identification of the processor during initialization of the processor (e.g., col., 27, lines 13 – 28).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Browning-IBM, O'Shea-Intel, Endo-Hitachi, Stracovsky-Infineon, Rose-IBM and Nota-Hitachi with the teachings of Colley – NCUBE in order to facilitate determining the identification of the processor during initialization of the processor because the identification of the processor would help select the processor for processing information. The well-known concept of using a processor for initialization would help initialize the processor (itself) and the system.

Conclusion

The prior art made of record (forms PTO-892 and applicant provided IDS cited arts) and not relied upon is considered pertinent to applicant's disclosure.

Ronkka et al., 6,631,394, also teaches the concept of handling exceptions/interrupts using common interrupt/exception handler for different types of operating systems.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Haresh Patel whose telephone number is (703) 605-5234. The examiner can normally be reached on Monday, Tuesday, Thursday and Friday from 10:00 am to 8:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Follansbee, can be reached at (703) 305-8498.

The appropriate fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Haresh Patel

~~July 9, 2004~~

2/15/05

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100